Bayesian Models in Machine Learning

Approximate inference in Bayesian models
Latent Dirichlet Allocation Model

Lukáš Burget

BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Latent Dirichlet Allocation Model

Set of topic (color) specific word distributions

Document specific topic mixture weights

$$\boldsymbol{\Phi}^{\mathrm{T}} = \begin{bmatrix} \boldsymbol{\varphi}_1^T \\ \boldsymbol{\varphi}_2^T \\ \boldsymbol{\varphi}_3^T \end{bmatrix} = \begin{array}{c} \begin{matrix} tenis & surfing & software & apple & burger & \\ \end{matrix} \\ \begin{bmatrix} 0.4 & 0.2 & 0.0 & 0.0 & 0.0 & \ldots \\ 0.0 & 0.4 & 0.2 & 0.2 & 0.0 & \ldots \\ 0.0 & 0.0 & 0.0 & 0.6 & 0.2 & \ldots \end{bmatrix} \end{array} \begin{matrix} sports \\ computers \\ food \end{matrix} \quad \begin{bmatrix} 0.5 \\ 0.0 \\ 0.5 \end{bmatrix} = \boldsymbol{\theta}_d$$

$$\boldsymbol{\Phi}\boldsymbol{\theta}_d = \begin{bmatrix} 0.2 & 0.1 & 0.0 & 0.3 & 0.1 & \ldots \end{bmatrix}$$

$$\boldsymbol{w}_d = [apple \quad burger \quad is \quad surfing \quad apple \quad tenis\,]$$

Document specific word distribution.

Mixture of Categorical distributions is again Categorical distribution.

Document, where each word is independently drawn from the document specific distribution

$$w_{jt} \sim \mathrm{Cat}(\boldsymbol{\Phi}\boldsymbol{\theta}_d)$$

The Task

Given a set of documents and chosen number of topics

learn the set of topic specific word distribution

without knowing any topic labels (i.e. topics are latent)

and, for each document, estimate the topic mixture weights.

$$\boldsymbol{\Phi}^{\mathrm{T}} = \begin{bmatrix} \boldsymbol{\varphi}_1^T \\ \boldsymbol{\varphi}_2^T \\ \boldsymbol{\varphi}_3^T \end{bmatrix} = \begin{array}{c} \begin{array}{cccccc} \mathit{tenis} & \mathit{surfing} & \mathit{software} & \mathit{apple} & \mathit{burger} & \\ \end{array} \\ \begin{bmatrix} 0.4 & 0.2 & 0.0 & 0.0 & 0.0 & \ldots \\ 0.0 & 0.4 & 0.2 & 0.2 & 0.0 & \ldots \\ 0.0 & 0.0 & 0.0 & 0.6 & 0.2 & \ldots \end{bmatrix} \end{array} \begin{array}{c} \mathit{sports} \\ \mathit{computers} \\ \mathit{food} \end{array} \quad \begin{bmatrix} 0.5 \\ 0.0 \\ 0.5 \end{bmatrix} = \boldsymbol{\theta}_d$$

This vector can serve as low-dimensional representation of the document (e.g. for topic clustering).
Group of documents for which the same weight dominates are probably on one and the same topic.

$$\boldsymbol{\Phi}\boldsymbol{\theta}_d = [0.2 \quad 0.1 \quad 0.0 \quad 0.3 \quad 0.1 \quad \ldots]$$

$$\begin{aligned} \boldsymbol{w}_1 &= [\mathit{apple} \quad \mathit{burger} \quad \mathit{is} \quad \mathit{surfing} \quad \mathit{apple} \quad \mathit{tenis}] \\ \boldsymbol{w}_2 &= [\mathit{tenis} \quad \mathit{and} \quad \mathit{surfing} \quad \mathit{but} \quad \mathit{mainly} \quad \mathit{tenis}] \\ \boldsymbol{w}_3 &= [\mathit{surfing} \quad \mathit{with} \quad \mathit{apple} \quad \mathit{sotware} \quad \mathit{about} \quad \mathit{tenis}] \\ &\vdots \\ \boldsymbol{w}_D &= [\mathit{the} \quad \mathit{best} \quad \mathit{burget} \quad \mathit{is} \quad \mathit{appe} \quad \mathit{burger}] \end{aligned}$$

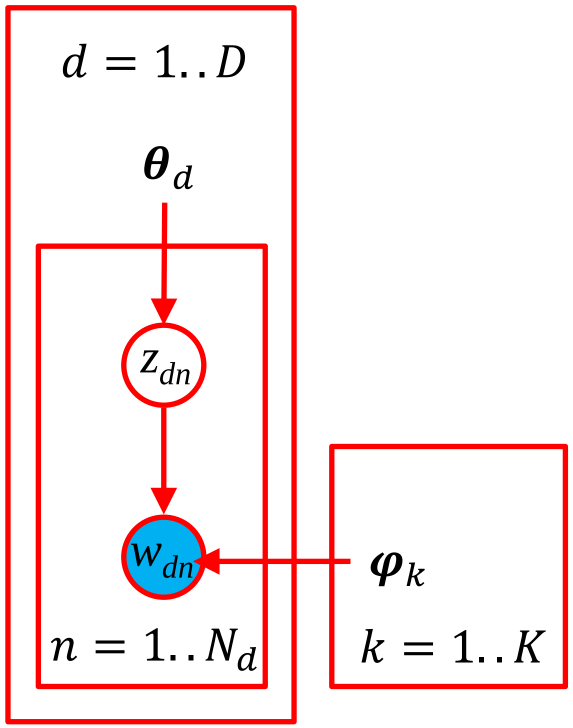
LDA assumed generative process

- For each document d, each word w_{dn} is independently drawn from the document specific distribution $\mathbf{\Phi}\boldsymbol{\theta}_d$:

$$
\begin{aligned}
&\text{for } d = 1..D \\
&\quad \text{for } n = 1..N_d \\
&\quad\quad w_{dn} \sim \mathrm{Cat}(\mathbf{\Phi}\boldsymbol{\theta}_d)
\end{aligned}
$$

- Or, like for GMM, we choose the mixture component z_{dn} (representing a topic) and sample observation w_{dn} from its distribution.

$$
\begin{aligned}
&\text{for } d = 1..D \\
&\quad \text{for } n = 1..N_d \\
&\quad\quad z_{dn} \sim \mathrm{Cat}(\boldsymbol{\theta}_d) \\
&\quad\quad w_{dn} \sim \mathrm{Cat}\left(\boldsymbol{\varphi}_{z_{dn}}\right)
\end{aligned}
$$

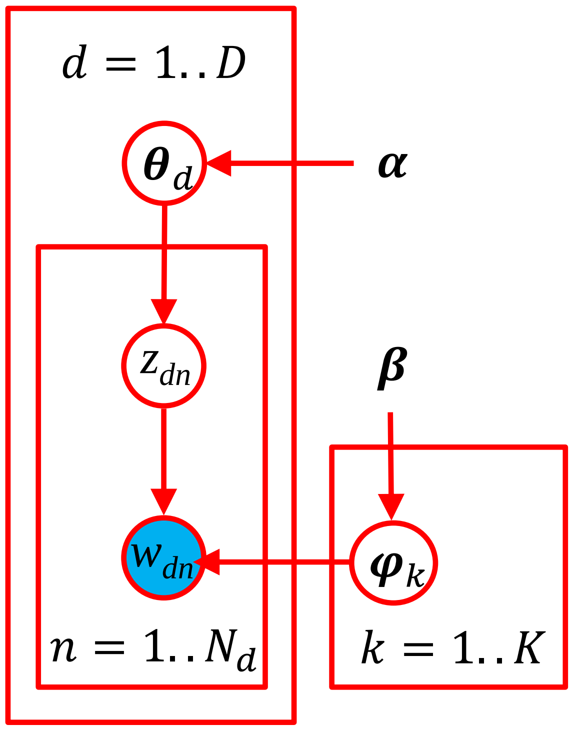


Full Bayesian LDA

- Let's treat parameters of each topic specific distribution $\boldsymbol{\varphi}_k$ and document specific weights $\boldsymbol{\theta}_d$ as random variables with Dirichlet priors: $\boldsymbol{\varphi}_k \sim \mathrm{Dir}(\boldsymbol{\beta})$ and $\boldsymbol{\theta}_d \sim \mathrm{Dir}(\boldsymbol{\alpha})$

- The generative process is now:

$$
\begin{aligned}
&\text{for } k = 1..K \\
&\quad \boldsymbol{\varphi}_{\boldsymbol{k}} \sim \mathrm{Dir}(\boldsymbol{\beta}) \\
&\text{for } d = 1..D \\
&\quad \boldsymbol{\theta}_d \sim \mathrm{Dir}(\boldsymbol{\alpha}) \\
&\quad \text{for } n = 1..N_d \\
&\qquad z_{dn} \sim \mathrm{Cat}(\boldsymbol{\theta}_d) \\
&\qquad w_{dn} \sim \mathrm{Cat}\left(\boldsymbol{\varphi}_{z_{dn}}\right)
\end{aligned}
$$



Bayesian LDA model summary

Joint Probability:

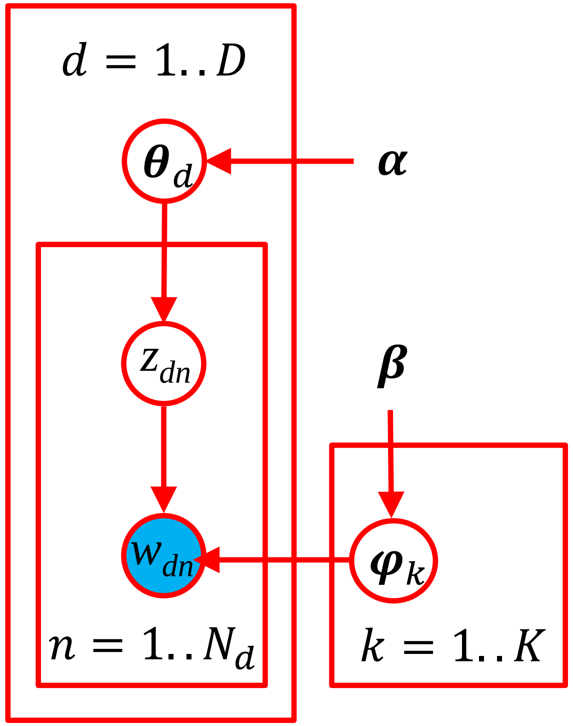
$$p(\boldsymbol{W},\boldsymbol{Z},\boldsymbol{\Theta},\boldsymbol{\Phi}) = \left(\prod_{k=1}^{K} p(\boldsymbol{\varphi}_k)\right)\left(\prod_{d=1}^{D} p(\boldsymbol{\theta}_d)\right)\left(\prod_{d=1}^{D}\prod_{n=1}^{N_d} P(z_{dn}|\boldsymbol{\theta}_d)\right)\left(\prod_{d=1}^{D}\prod_{n=1}^{N_d} P(w_{dn}|\boldsymbol{\Phi}, z_{dn})\right)$$

Variables:

- $\boldsymbol{\Phi} = [\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, \dots, \boldsymbol{\varphi}_K]$ - topic specific word distributions
- $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_D]$ - document specific topic distributions
- $z_{dn} = k$ - denotes that n^{th} word in document d comes from topic k
- $w_{dn} = v$ - denotes that n^{th} word in document d is v

Indices:

- $d = 1..D$ - (training) document
- $k = 1..K$ - topic
- $v = 1..V$ - unique word in the vocabulary of size V
- $n = 1..N_d$ - position of word in document d



$p(\boldsymbol{\varphi}_k) = \mathrm{Dir}(\boldsymbol{\varphi}_k|\boldsymbol{\beta}) \propto \prod_{v=1}^{V} \varphi_{kv}^{\beta_v - 1}$ - prior on parameters $\boldsymbol{\varphi}_k$

$p(\boldsymbol{\theta}_d) = \mathrm{Dir}(\boldsymbol{\theta}_d|\boldsymbol{\alpha}) \propto \prod_{k=1}^{K} \theta_{dk}^{\alpha_k - 1}$ - prior on document specific topic mixture weights

$P(z_{dn} = k|\boldsymbol{\theta}_d) = \theta_{dk}$ - probability that a word in document d comes from topic k

$P(w_{dn} = v|\boldsymbol{\Phi}, z_{dn} = k) = \varphi_{kv}$ - probability of word v if we know that it comes from topic k

Using counts $C_{d,v}^k$

Joint Probability:

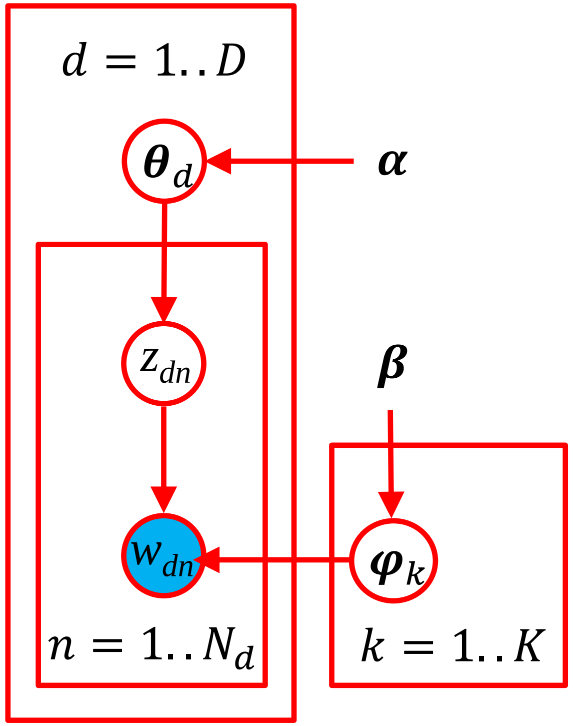
$$p(\boldsymbol{W},\boldsymbol{Z},\boldsymbol{\Theta},\boldsymbol{\Phi}) = \left(\prod_{k=1}^{K} p(\boldsymbol{\varphi}_k)\right)\left(\prod_{d=1}^{D} p(\boldsymbol{\theta}_d)\right)\left(\prod_{d=1}^{D}\prod_{n=1}^{N_d} P(z_{dn}|\boldsymbol{\theta}_d)\right)\left(\prod_{d=1}^{D}\prod_{n=1}^{N_d} P(w_{dn}|\boldsymbol{\Phi},z_{dn})\right)$$

Let $C_{d,v}^k$ be the count of words v generated from topic k in document d as assigned by latent variables z_{dn}.

$$C_{d,v}^k = \sum_{n=1}^{N_d} \delta(z_{dn}=k)\delta(w_{dn}=v)$$

$$\prod_{n=1}^{N_d} P(z_{dn}|\boldsymbol{\theta}_d) = \prod_{n=1}^{N_d} \theta_{dz_{dn}} = \prod_{k=1}^{K} \theta_{dk}^{C_{d,(\cdot)}^k},$$

where $C_{d,(\cdot)}^k = \sum_{v=1}^{V} C_{d,v}^k$ is the count of all words from topic k in document d



$$\prod_{d=1}^{D}\prod_{n=1}^{N_d} P(w_{dn}|\boldsymbol{\Phi},z_{dn}) = \prod_{d=1}^{D}\prod_{n=1}^{N_d} \varphi_{z_{dn}w_{dn}} = \prod_{k=1}^{K}\prod_{v=1}^{V} \varphi_{kv}^{C_{(\cdot),v}^k},$$

where $C_{(\cdot),v}^k = \sum_{d=1}^{D} c_{d,v}^k$ is the count of words v from topic topic k in all documents.

Joint probability using the counts

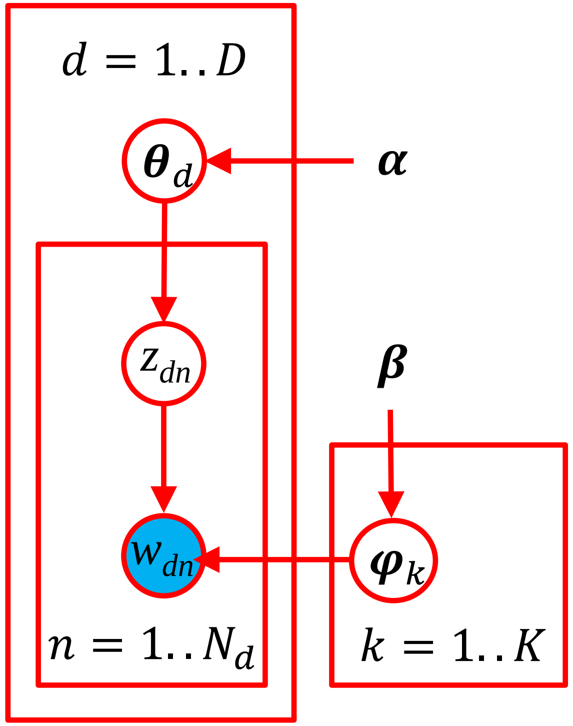
$$p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}) = \left(\prod_{k=1}^{K} p(\boldsymbol{\varphi}_k)\right)\left(\prod_{d=1}^{D} p(\boldsymbol{\theta}_d)\right)\left(\prod_{d=1}^{D}\prod_{n=1}^{N_d} P(z_{dn}|\boldsymbol{\theta}_d)\right)\left(\prod_{d=1}^{D}\prod_{n=1}^{N_d} P(w_{dn}|\boldsymbol{\Phi}, z_{dn})\right)$$

$$= \left(\prod_{k=1}^{K} \frac{\Gamma\left(\sum_{v=1}^{V}\beta_v\right)}{\prod_{v=1}^{V}\Gamma(\beta_v)} \prod_{v=1}^{V} \varphi_{kv}^{\beta_v - 1}\right)\left(\prod_{d=1}^{D} \frac{\Gamma\left(\sum_{k=1}^{K}\alpha_k\right)}{\prod_{k=1}^{K}\Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_{dk}^{\alpha_k - 1}\right)\left(\prod_{d=1}^{D}\prod_{k=1}^{K} \theta_{dk}^{c_{d,(\cdot)}^{k}}\right)\left(\prod_{k=1}^{K}\prod_{v=1}^{V} \varphi_{kv}^{c_{(\cdot),v}^{k}}\right)$$

$$\propto \left(\prod_{k=1}^{K}\prod_{v=1}^{V} \varphi_{kv}^{\beta_v - 1}\right)\left(\prod_{d=1}^{D}\prod_{k=1}^{K} \theta_{dk}^{\alpha_k - 1}\right)\left(\prod_{d=1}^{D}\prod_{k=1}^{K} \theta_{dk}^{c_{d,(\cdot)}^{k}}\right)\left(\prod_{k=1}^{K}\prod_{v=1}^{V} \varphi_{kv}^{c_{(\cdot),v}^{k}}\right)$$

$$c_{d,v}^{k} = \sum_{n=1}^{N_d} \delta(z_{dn} = k)\delta(w_{dn} = v)$$

$$c_{d,(\cdot)}^{k} = \sum_{v=1}^{V} c_{d,v}^{k} \qquad c_{(\cdot),v}^{k} = \sum_{d=1}^{D} c_{d,v}^{k}$$



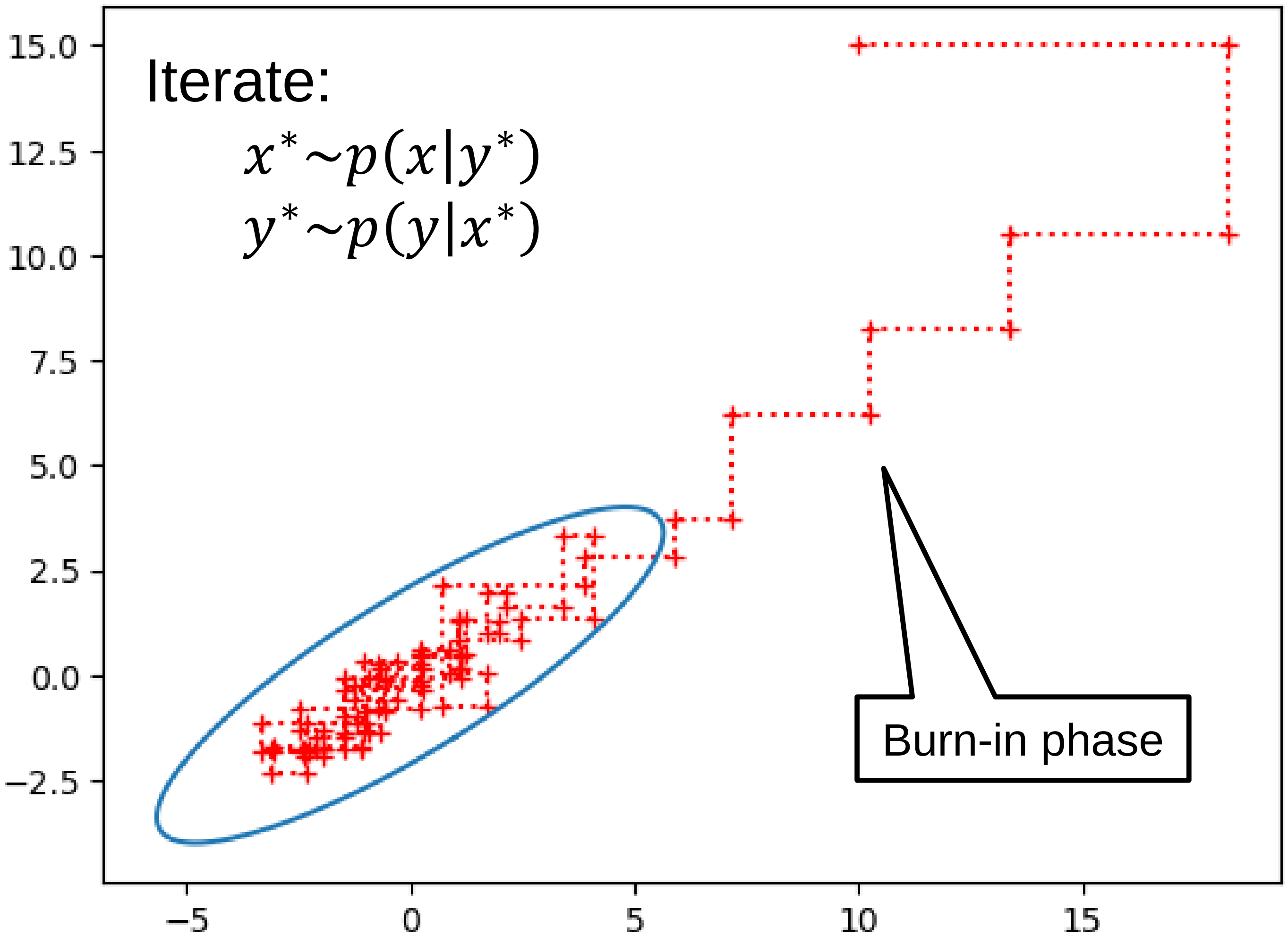
Gibbs Sampling

- Assume we cannot sample from the complex joint distribution $p(z_1, z_2)$ but it is possible to sample from the conditional distributions $p(z_1|z_2)$ and $p(z_2|z_1)$
  1. Initialize z_1^* to any value (i.e. chosen constant)
  2. Given current sample z_1^* generate $\mathrm{z}_2^* \sim p(z_2|z_1)$
  3. Given current sample z_2^* generate $\mathrm{z}_1^* \sim p(z_1|z_2)$
  4. Go to steps 2.

- In theory, after infinite number of iteration the final values z_1^*, z_2^* is a sample from $p(z_1, z_2)$
- Or, with increasing number of iterations, z_1^*, z_2^* converges to a valid sample from $p(z_1, z_2)$
- In practice, after several initial iterations (burn-in phase) take z_1^*, z_2^* from every N^{th} iteration and consider them samples from $p(z_1, z_2)$
  - Often $N = 1$ is used
  - Starting from a likely value of z_1^* requires less burn-in iterations
- This can be extended to any number variables
  - always sample one given current values for others
- Works for any random variables (discrete, continuous; scalars, vectors)

Gibbs sampling for 2D Gaussian

Of course, it is possible to efficiently and exactly sample directly from a 2D Gaussian distribution. We use this toy example only to demonstrate how Gibbs sampling works.



For 2D gaussian distribution

$$p\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix} x \\ y \end{bmatrix} \middle\| \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{xx} \end{bmatrix}\right)$$

the conditional probability

$$p(x|y) = \mathcal{N}\left(x \middle| \mu_{x|y}, \Lambda_{xx}^{-1}\right)$$

where

$$\mu_{x|y} = \mu_x - \Lambda_{xx}^{-1}\Lambda_{xy}(y - \mu_y)$$

and

$$\begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{xx} \end{bmatrix}^{-1} = \begin{bmatrix} \Lambda_{xx} & \Lambda_{xy} \\ \Lambda_{yx} & \Lambda_{xx} \end{bmatrix}$$

Approximate inference (for Bayesian LDA)

- Gibbs sampling
  - Instead of obtaining $p(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}|\boldsymbol{W})$, we only generate samples from this distribution
  - We alternately sample from
    - » $\boldsymbol{Z}^{*} \sim \mathrm{P}(\boldsymbol{Z}|\ \boldsymbol{\Theta}, \boldsymbol{\Phi}, \boldsymbol{W})$
    - » $\boldsymbol{\Theta}^{*}, \boldsymbol{\Phi}^{*} \sim \mathrm{P}(\boldsymbol{\Theta}, \boldsymbol{\Phi}|\boldsymbol{Z}, \boldsymbol{W})$

- Variational Bayes
  - Approximate intractable $p(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}|\boldsymbol{W})$ with tractable $q(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi})$
  - Iteratively tune parameters of$q(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi})$ to minimize $D_{KL}\ (q(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi})||p(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}|\boldsymbol{W}))$

- …

GS inference for LDA: Sampling $\boldsymbol{Z}$

$$p(\boldsymbol{W},\boldsymbol{Z},\boldsymbol{\Theta},\boldsymbol{\Phi}) = \left(\prod_{k=1}^{K} p(\boldsymbol{\varphi}_k)\right)\left(\prod_{d=1}^{D} p(\boldsymbol{\theta}_d)\right)\left(\prod_{d=1}^{D}\prod_{n=1}^{N_d} P(z_{dn}|\boldsymbol{\theta}_d)\right)\left(\prod_{d=1}^{D}\prod_{n=1}^{N_d} P(w_{dn}|\boldsymbol{\Phi}, z_{dn})\right)$$

Step 1:

- sample $\boldsymbol{Z}^* \sim \mathrm{P}(\boldsymbol{Z}|\,\boldsymbol{\Theta},\boldsymbol{\Phi},\boldsymbol{W})$
- or equivalently for each d and n sample $z_{dn}^* \sim P(z_{dn}|\boldsymbol{\Theta},\boldsymbol{\Phi},\boldsymbol{W})$

We want posterior as a function of $\boldsymbol{Z}$, so we select only the terms involving $\boldsymbol{Z}$. The other terms are constant.

$$\mathrm{P}(\boldsymbol{Z}|\,\boldsymbol{\Theta},\boldsymbol{\Phi},\boldsymbol{W}) \propto P(\boldsymbol{W},\boldsymbol{Z},\boldsymbol{\theta},\boldsymbol{\Phi}) \propto \prod_{d=1}^{D}\prod_{n=1}^{N_d} P(z_{dn}|\boldsymbol{\theta}_d)\, P(w_{dn}|\boldsymbol{\Phi}, z_{dn}) \propto \prod_{d=1}^{D}\prod_{n=1}^{N_d} P(z_{dn}|\boldsymbol{\Theta},\boldsymbol{\Phi},\mathbf{W})$$

$$P(z_{dn}|\boldsymbol{\Theta},\boldsymbol{\Phi},\boldsymbol{W}) \propto {\color{red}P(w_{dn}|\boldsymbol{\Phi}, z_{dn})}{\color{blue}P(z_{dn}|\boldsymbol{\theta}_d)} = {\color{red}\varphi_{z_{dn}w_{dn}}}{\color{blue}\theta_{dz_{dn}}}$$

$$P(z_{dn}|\boldsymbol{\Theta},\boldsymbol{\Phi},\boldsymbol{W}) = \frac{\varphi_{z_{dn}w_{dn}}\theta_{dz_{dn}}}{\sum_{k'=1}^{K}\varphi_{k'w_{dn}}\theta_{dk'}} = \pi_{dw_{dn}z_{dn}}$$

$$\pi_{dvk} = \frac{\varphi_{kv}\theta_{dk}}{\sum_{k'=1}^{K}\varphi_{k'v}\theta_{dk'}}$$

Factorizes into product of independent terms one for each z_{dn} $\Rightarrow$ each z_{dn} can be sampled from independent $P(z_{dn}|\boldsymbol{\Theta},\boldsymbol{\Phi},\mathbf{W})$

GS inference for LDA: Sampling $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$

$$p(\boldsymbol{W},\boldsymbol{Z},\boldsymbol{\Theta},\boldsymbol{\Phi}) \propto \left(\prod_{k=1}^{K}\prod_{v=1}^{V}\varphi_{kv}^{\beta_v-1}\right)\left(\prod_{d=1}^{D}\prod_{k=1}^{K}\theta_{dk}^{\alpha_k-1}\right)\left(\prod_{d=1}^{D}\prod_{k=1}^{K}\theta_{dk}^{C_{d,(\cdot)}^{k}}\right)\left(\prod_{k=1}^{K}\prod_{v=1}^{V}\varphi_{kv}^{C_{(\cdot),v}^{k}}\right)$$

Step 2:

- sample $\boldsymbol{\Theta}^*,\boldsymbol{\Phi}^*\sim\mathrm{P}(\boldsymbol{\Theta},\boldsymbol{\Phi}|\boldsymbol{Z},\boldsymbol{W})$
- or equivalently for each $d=1..D$ sample $\boldsymbol{\theta}_d^*\sim P(\boldsymbol{\theta}_d|\boldsymbol{Z},\boldsymbol{W})$
  and for each $k=1..K$ sample $\boldsymbol{\varphi}_k^*\sim P(\boldsymbol{\varphi}_k|\boldsymbol{Z},\boldsymbol{W})$

Factorizes into product of independent terms for each $\boldsymbol{\theta}_d$ and $\boldsymbol{\varphi}_k \Rightarrow$ can be sampled independently.

The dependence on $\boldsymbol{Z}$ is through the counts $C_{d,v}^{k}$

$$\mathrm{P}(\boldsymbol{\Theta},\boldsymbol{\Phi}|\boldsymbol{Z},\boldsymbol{W}) \propto P(\boldsymbol{W},\boldsymbol{Z},\boldsymbol{\theta},\boldsymbol{\Phi}) \propto \left(\prod_{d=1}^{D}\prod_{k=1}^{K}\theta_{dk}^{\alpha_k+C_{d,(\cdot)}^{k}-1}\right)\left(\prod_{k=1}^{K}\prod_{v=1}^{V}\varphi_{kv}^{\beta_v+C_{(\cdot),v}^{k}-1}\right)$$

$$\mathrm{P}(\boldsymbol{\Theta},\boldsymbol{\Phi}|\boldsymbol{Z},\boldsymbol{W}) = \prod_{d=1}^{D}\mathrm{Dir}\left(\boldsymbol{\theta}_d|\boldsymbol{\alpha}+\boldsymbol{C}_{d,(\cdot)}\right)\prod_{k=1}^{K}\mathrm{Dir}\left(\boldsymbol{\varphi}_k|\boldsymbol{\beta}+\boldsymbol{C}_{(\cdot)}^{k}\right) = \prod_{d=1}^{D}P(\boldsymbol{\theta}_d|\boldsymbol{Z},\boldsymbol{W})\prod_{k=1}^{K}P(\boldsymbol{\varphi}_k|\boldsymbol{Z},\boldsymbol{W})$$

where vectors $\boldsymbol{C}_{d,(\cdot)} = \left[C_{d,(\cdot)}^{1}, C_{d,(\cdot)}^{2}, \dots, C_{d,(\cdot)}^{K}\right]^T$ and $\boldsymbol{C}_{(\cdot)}^{k} = \left[C_{(\cdot),1}^{k}, C_{(\cdot),2}^{k}, \dots, C_{(\cdot),V}^{k}\right]^T$

GS inference for LDA: Using word counts

- Until now, each training document $d = 1..D$ was represented by a variable length sequence of N_d words w_{dn}, where $n = 1..N_d$
- More conveniently, we can represent all documents by $D \times V$ matrix $\boldsymbol{M}$, with elements M_{dv} counting how many times document d contains word v
  - No need to know the order of the words w_{dn} in the sequence. The counts M_{dv} are enough.
  - Instead of sampling each z_{dn}, directly sample $C_{d,v}^k$

$$\pi_{dvk} = \frac{\varphi_{kv}\theta_{dk}}{\sum_{k'=1}^{K} \varphi_{k'v}\theta_{dk'}}$$

$$\boldsymbol{\pi}_{dv} = [\pi_{dv1}, \pi_{dv2}, \dots, \pi_{dvK}]^T$$

for $n = 1..N_d$

$$z_{dn}^* \sim P(z_{dn}|\boldsymbol{\Theta}, \boldsymbol{\Phi}, \boldsymbol{W}) = \mathrm{Cat}\left(\boldsymbol{\pi}_{dw_{dn}}\right)$$

$$C_{d,v}^k = \sum_{n=1}^{N_d} \delta(z_{dn} = k)\delta(w_{dn} = v)$$

$$\boldsymbol{C}_{d,v} = \begin{bmatrix} C_{d,v}^1 \\ C_{d,v}^2 \\ \vdots \\ C_{d,v}^K \end{bmatrix}$$

$$\boldsymbol{C}_{d,v} \sim \mathrm{Multinomial}(\boldsymbol{\pi}_{dv}, M_{dv})$$

GS inference for LDA: Summary

for number of GS iterations

for $d = 1..D$

for $v = 1..V$

$\boldsymbol{C}_{d,v} \sim \text{Multinomial}(\boldsymbol{\pi}_{dv}, M_{dv})$

for $d = 1..D$

$\boldsymbol{\theta}_d^* \sim P(\boldsymbol{\theta}_d | \boldsymbol{Z}, \boldsymbol{W}) = \text{Dir}\left(\boldsymbol{\theta}_d | \boldsymbol{\alpha} + \boldsymbol{C}_{d,(\cdot)}\right)$

for $k = 1..K$

$\boldsymbol{\varphi}_k^* \sim P(\boldsymbol{\varphi}_k | \boldsymbol{Z}, \boldsymbol{W}) = \text{Dir}\left(\boldsymbol{\varphi}_k | \boldsymbol{\beta} + \boldsymbol{C}_{(\cdot)}^k\right)$

where

$$C_{d,(\cdot)}^k = \sum_{v=1}^{V} C_{d,v}^k, \qquad C_{(\cdot),\boldsymbol{v}}^k = \sum_{d=1}^{D} C_{d,v}^k$$

$$\boldsymbol{C}_{d,(\cdot)} = \begin{bmatrix} C_{d,(\cdot)}^1 \\ C_{d,(\cdot)}^2 \\ \vdots \\ C_{d,(\cdot)}^K \end{bmatrix}, \qquad \boldsymbol{C}_{(\cdot)}^k = \begin{bmatrix} C_{(\cdot),1}^k \\ C_{(\cdot),2}^k \\ \vdots \\ C_{(\cdot),V}^k \end{bmatrix}$$

Where $\boldsymbol{\pi}_{dv}$ is evaluated using the currently values of $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$ in each iteration.

After running a number of GS iterations, **we get likely samples** (from the posterior $p(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi} | \boldsymbol{W})$) of vectors $\boldsymbol{\theta}_d$ representing each document and $\boldsymbol{\varphi}_\mathrm{k}$ representing latent topic distributions.

Approximate inference
(for Bayesian LDA)

- Gibbs sampling
  - Instead of obtaining $p(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}|\boldsymbol{W})$, we only generate samples from this distribution

- Variational Bayes
  - Approximate intractable $p(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}|\boldsymbol{W})$ with tractable $q(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi})$
  - Iteratively tune parameters of$q(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi})$ to minimize $D_{KL}\ (q(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi})||p(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}|\boldsymbol{W}))$

- …

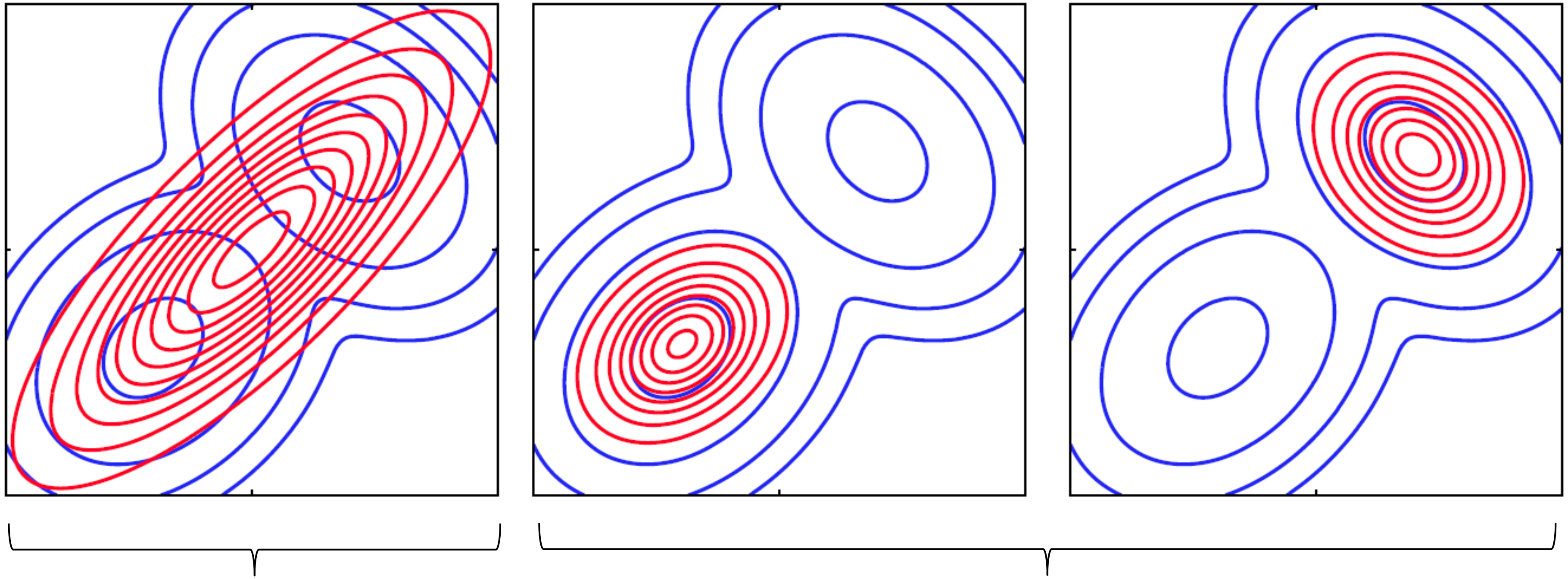
Variational Bayes

$$\ln p(\mathbf{X}) = \underbrace{\int q(\mathbf{Y}) \ln p(\mathbf{X}, \mathbf{Y}) \, \mathrm{d}\mathbf{Y} - \int q(\mathbf{Y}) \ln q(\mathbf{Y}) \, \mathrm{d}\mathbf{Y}}_{\mathcal{L}(q(\mathbf{Y}))} \underbrace{- \int q(\mathbf{Y}) \ln \frac{p(\mathbf{Y}|\mathbf{X})}{q(\mathbf{Y})} \, \mathrm{d}\mathbf{Y}}_{D_{KL}(q(\mathbf{Y})||p(\mathbf{Y}|\mathbf{X}))}$$

- Find $q(\mathbf{Y})$, which is a good approximation for the true posterior $p(\mathbf{Y}|\mathbf{X})$
- Maximize $\mathcal{L}\big(q(\mathbf{Y})\big)$ w.r.t. $q(\mathbf{Y})$, which in turn minimizes $D_{KL}\big(q(\mathbf{Y})\big|\big|p(\mathbf{Y}|\mathbf{X})\big)$
  - “Handcraft” a reasonable parametric distribution $q(\mathbf{Y}|\boldsymbol{\eta})$ and optimize $\mathcal{L}\big(q(\mathbf{Y}|\boldsymbol{\eta})\big)$ w.r.t. its parameters $\boldsymbol{\eta}$.
  - Mean field approximation assuming factorized form $q(\mathbf{Y})$=$q(\mathbf{Y}_1)q(\mathbf{Y}_2)q(\mathbf{Y}_3)$…

Minimizing Kullback-Leibler divergence

- We optimize parameters of (simpler) distribution $q(\mathbf{Y})$ to minimize Kullback-Leibler divergence between $q(\mathbf{Y})$ and $p(\mathbf{Y}|\mathbf{X})$.



- Minimizing $D_{KL}\big(p(\mathbf{Y}|\mathbf{X})||q(\mathbf{Y})\big)$.
- Not VB objective
- Expectation propagation

- Two local optima when (numerically) minimizing $D_{KL}\big(q(\mathbf{Y})||p(\mathbf{Y}|\mathbf{X})\big)$.
- VB performs this optimization

VB – Mean field approximation

- Popular Variational Bayes optimization method
- Variant of Variational Bayes, where the set of model variables $\mathbf{Y}$, can be split into subsets $\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3$, ..., with conditionally conjugate priors
  - $p(\mathbf{Y}_\mathrm{i}|\mathbf{X}, \mathbf{Y}_{\forall \mathrm{j} \neq \mathrm{i}})$ is tractable with conjugate prior
  - E.g. for Bayesian GMM $p(\mu_c, \lambda_c|\mathbf{X}, \mathbf{z})$ has $\mathrm{NormalGamma}$ prior
- We assume factorized approximate posterior

$$q(\mathbf{Y}) = q(\mathbf{Y}_1)q(\mathbf{Y}_2)q(\mathbf{Y}_3)\ldots = \prod_{\boldsymbol{i}} q(\mathbf{Y}_\mathrm{i})$$

- This factorization dictates the optimal (conjugate) distributions for the factors $q(\mathbf{Y}_\mathrm{i})$ and brings well defined iterative update formulas:

$$q(\mathbf{Y}_i)^* \propto \exp\left(\int q(\mathbf{Y}_{\forall \mathrm{j} \neq \mathrm{i}}) \ln p(\mathbf{X}, \mathbf{Y}) \, \mathrm{d}\mathbf{Y}_{\forall \mathrm{j} \neq \mathrm{i}}\right)$$

Mean field - update

$$\mathcal{L}(q(\mathbf{Y})) = \int q(\mathbf{Y}) \ln p(\mathbf{X}, \mathbf{Y})\ d\mathbf{Y} - \int q(\mathbf{Y}) \ln q(\mathbf{Y})\ d\mathbf{Y} = \int \prod_{i=1}^{M} q(\mathbf{Y}_i)\ \left[\ln\ p(\mathbf{X}, \mathbf{Y}) - \ln \prod_i\ q(\mathbf{Y}_i)\right] d\mathbf{Y}$$

$$= \int \prod_{i=1}^{M} q(\mathbf{Y}_i)\ \left[\ln\ p(\mathbf{X}, \mathbf{Y}) - \sum_i \ln\ q(\mathbf{Y}_i)\right]\ d\mathbf{Y}$$

- For example, let $M = 3$
- Now, lets optimize the lower bound $\mathcal{L}(q(\mathbf{Y}_1))$ w.r.t only one distribution $q(\mathbf{Y}_1)$

$$\mathcal{L}(q(\mathbf{Y}_1)) = \iiint q(\mathbf{Y}_1)q(\mathbf{Y}_2)q(\mathbf{Y}_3)\ [\ln p(\mathbf{X}, \mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3) - \ln q(\mathbf{Y}_1) - \ln q(\mathbf{Y}_2) - \ln q(\mathbf{Y}_3)]\ d\mathbf{Y}_1\ d\mathbf{Y}_2\ d\mathbf{Y}_3$$

$$= \int q(\mathbf{Y}_1) \underbrace{\iint q(\mathbf{Y}_2)q(\mathbf{Y}_3)\ \ln p(\mathbf{X}, \mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3)\ d\mathbf{Y}_2\ d\mathbf{Y}_3}_{\ln \tilde{p}(\mathbf{Y}_1) + const}\ d\mathbf{Y}_1 - \int q(\mathbf{Y}_1) \ln q(\mathbf{Y}_1)\ d\mathbf{Y}_1 + const$$

$$= \int q(\mathbf{Y}_1) \ln \tilde{p}(\mathbf{Y}_1)\ d\mathbf{Y}_1 - \int q(\mathbf{Y}_1) \ln q(\mathbf{Y}_1)\ d\mathbf{Y}_1 + const = -D_{KL}(q(\mathbf{Y}_1)||\tilde{p}(\mathbf{Y}_1)) + const$$

where $\tilde{p}(\mathbf{Y}_1)$ is normalized to be a valid distribution (therefore $+const$)

- $\mathcal{L}(q(\mathbf{Y}_1))$ is maximized by setting the D_{KL} term to zero, which implies

$$\ln q(\mathbf{Y}_1) = \ln \tilde{p}(\mathbf{Y}_1) = \iint q(\mathbf{Y}_2)q(\mathbf{Y}_3)\ \ln p(\mathbf{X}, \mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3)\ d\mathbf{Y}_2\ d\mathbf{Y}_3 + const$$

- In general, we can iteratively update each $q(\mathbf{Y}_\mathrm{i})$ given the others $q(\mathbf{Y}_{\mathrm{i}\neq j})$ as:

$$q(\mathbf{Y}_j) \propto \exp \int q(\mathbf{Y}_{\forall j \neq i}) \ln p(\mathbf{X}, \mathbf{Y})\ d\mathbf{Y}_{\forall j \neq i}$$

where each update guaranties to improve the lower bound $\mathcal{L}(q(\mathbf{Y}))$

Variational Bayes for LDA

Variational Bayes updates dictate:

$$q(\mathbf{Y}_i)^* \propto \exp \int q(\mathbf{Y}_{\forall \mathrm{j} \neq \mathrm{i}}) \ln p(\mathbf{X}, \mathbf{Y}) \, \mathrm{d}\mathbf{Y}_{\forall \mathrm{j} \neq \mathrm{i}}$$

For the LDA model we chose to approximate the posterior using factorization

$$p(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi} | \boldsymbol{W}) \approx q(\boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}) = q(\boldsymbol{Z}) q(\boldsymbol{\Theta}, \boldsymbol{\Phi})$$

Therefore, we search for updates in form:

$$q(\boldsymbol{Z})^* \propto \exp \iint q(\boldsymbol{\Theta}, \boldsymbol{\Phi}) \ln p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}) \mathrm{d}\boldsymbol{\Theta} \mathrm{d}\boldsymbol{\Phi}$$

$$q(\boldsymbol{\Theta}, \boldsymbol{\Phi})^* \propto \exp \sum_{Z} q(\boldsymbol{Z}) \ln p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi})$$

Or equivalently:

$$\ln q(\boldsymbol{Z})^* = \mathbb{E}_{q(\boldsymbol{\Theta}, \boldsymbol{\Phi})} [\ln p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi})] + \mathrm{const}$$

$$\ln q(\boldsymbol{\Theta}, \boldsymbol{\Phi})^* = \mathbb{E}_{q(\boldsymbol{Z})} [\ln p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi})] + \mathrm{const}$$

VB update for $q(\boldsymbol{Z})$

$$p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}) = \left(\prod_{k=1}^{K} p(\boldsymbol{\varphi}_k)\right)\left(\prod_{d=1}^{D} p(\boldsymbol{\theta}_d)\right)\left(\prod_{d=1}^{D}\prod_{n=1}^{N_d} P(z_{dn}|\boldsymbol{\theta}_d)\right)\left(\prod_{d=1}^{D}\prod_{n=1}^{N_d} P(w_{dn}|\boldsymbol{\Phi}, z_{dn})\right)$$

$$\ln p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi}) = \sum_{k=1}^{K} \ln p(\boldsymbol{\varphi}_k) + \sum_{d=1}^{D}\left(\ln p(\boldsymbol{\theta}_d) + \sum_{n=1}^{N_d} \ln P(z_{dn}|\boldsymbol{\theta}_d) + \ln P(w_{dn}|\boldsymbol{\Phi}, z_{dn})\right)$$

The terms in $\ln p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi})$ independent of z_{dn} are absorbed in const_2

$$\begin{aligned}\ln q(\boldsymbol{Z})^* &= \mathbb{E}_{q(\boldsymbol{\Theta},\boldsymbol{\Phi})}[\ln p(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\Theta}, \boldsymbol{\Phi})] + \mathrm{const}_1 \\ &= \sum_{d=1}^{D}\sum_{n=1}^{N_d} \mathbb{E}_{q(\boldsymbol{\Theta},\boldsymbol{\Phi})}[\ln P(z_{dn}|\boldsymbol{\theta}_d) + \ln P(w_{dn}|\boldsymbol{\Phi}, z_{dn})] + \mathrm{const}_2 \\ &= \sum_{d=1}^{D}\sum_{n=1}^{N_d} \ln q(z_{dn})^* \quad \Rightarrow \quad q(\boldsymbol{Z})^* = \prod_{d=1}^{D}\prod_{n=1}^{N_d} q(z_{dn})^*\end{aligned}$$

- We only require factorization $q(\boldsymbol{Z})q(\boldsymbol{\Theta}, \boldsymbol{\Phi})$, but $q(\boldsymbol{Z})$ automatically further factorizes into a product of independent categorical distributions one for each z_{dn}- so called **induced factorization**

- We will derive the update for distributions $q(z_{dn})$ later.

VB update for $q(\mathbf{\Theta}, \mathbf{\Phi})$

$$\ln p(\boldsymbol{W}, \boldsymbol{Z}, \mathbf{\Theta}, \mathbf{\Phi}) = \sum_{k=1}^{K} \ln p(\boldsymbol{\varphi}_k) + \sum_{d=1}^{D} \left(\ln p(\boldsymbol{\theta}_d) + \sum_{n=1}^{N_d} \ln \theta_{d z_{dn}} + \ln \varphi_{z_{dn} w_{dn}} \right) + \text{const}$$

$$\begin{aligned}
\ln q(\mathbf{\Theta}, \mathbf{\Phi})^* &= \mathbb{E}_{q(\boldsymbol{Z})}[\ln p(\boldsymbol{W}, \boldsymbol{Z}, \mathbf{\Theta}, \mathbf{\Phi})] + \text{const} \\
&= \mathbb{E}_{q(\boldsymbol{Z})}\left[\sum_{d=1}^{D} \left(\ln p(\boldsymbol{\theta}_d) + \sum_{n=1}^{N_d} \ln \theta_{d z_{dn}} \right) + \left(\sum_{k=1}^{K} \ln p(\boldsymbol{\varphi}_k) + \sum_{d=1}^{D} \sum_{n=1}^{N_d} \ln \varphi_{z_{dn} w_{dn}} \right) \right] + \text{const} \\
&= \sum_{d=1}^{D} \left(\ln p(\boldsymbol{\theta}_d) + \sum_{n=1}^{N_d} \mathbb{E}_{q(z_{dn})}[\ln \theta_{d z_{dn}}] \right) + \sum_{k=1}^{K} \left(\ln p(\boldsymbol{\varphi}_k) + \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mathbb{E}_{q(z_{dn})}[\delta(z_{dn} = k) \ln \varphi_{k w_{dn}}] \right) + \text{const} \\
&= \sum_{d=1}^{D} \ln q(\boldsymbol{\theta}_d)^* + \sum_{k=1}^{K} \ln q(\boldsymbol{\varphi}_k)^* \quad \Rightarrow \quad q(\mathbf{\Theta}, \mathbf{\Phi})^* = \prod_{d=1}^{D} q(\boldsymbol{\theta}_d)^* \prod_{k=1}^{K} q(\boldsymbol{\varphi}_k)^*
\end{aligned}$$

induced factorization

VB update for $q(\boldsymbol{\theta}_d)$

$$\begin{aligned}
\ln q(\boldsymbol{\theta}_d)^* &= \ln \mathrm{Dir}(\boldsymbol{\theta}_d|\boldsymbol{\alpha}) + \sum_{n=1}^{N_d} \mathbb{E}_{q(z_{dn})}\left[\ln \theta_{dz_{dn}}\right] + \mathrm{const} \\
&= \left(\sum_{k=1}^{K} \alpha_k - 1 \ln \theta_{dk}\right) + \sum_{n=1}^{N_d} \sum_{k=1}^{K} q(z_{dn} = k) \ln \theta_{dk} + \mathrm{const} \\
&= \sum_{k=1}^{K} \left(\alpha_k - 1 + \sum_{n=1}^{N_d} q(z_{dn} = k)\right) \ln \theta_{dk} + \mathrm{const} \\
&= \sum_{k=1}^{K} \left(\alpha_k + \bar{C}_{d,(\cdot)}^{k} - 1\right) \ln \theta_{dk} + \mathrm{const} \\
&= \ln \mathrm{Dir}\left(\boldsymbol{\theta}_d|\boldsymbol{\alpha} + \overline{\boldsymbol{C}}_{d,(\cdot)}\right)
\end{aligned}$$

where

$$\bar{C}_{d,(\cdot)}^{k} = \sum_{n=1}^{N_d} q(z_{dn} = k)$$

$$\overline{\boldsymbol{C}}_{d,(\cdot)} = \left[\bar{C}_{d,(\cdot)}^{1}, \bar{C}_{d,(\cdot)}^{2}, \dots, \bar{C}_{d,(\cdot)}^{K}\right]$$

$$q(\boldsymbol{\theta}_d)^* = \mathrm{Dir}\left(\boldsymbol{\theta}_d|\boldsymbol{\alpha} + \overline{\boldsymbol{C}}_{d,(\cdot)}\right) = \mathrm{Dir}\left(\boldsymbol{\theta}_d|\boldsymbol{\alpha}_d^*\right)$$

Expected counts similar to the hard counts $C_{d,(\cdot)}^{k}$ from GS

Practically, the update means to calculate vector of parameters $\boldsymbol{\alpha}_d^* = \boldsymbol{\alpha} + \overline{\boldsymbol{C}}_{d,(\cdot)}$ for each d

VB update for $q(\boldsymbol{\varphi}_k)$

$$
\begin{aligned}
\ln q(\boldsymbol{\varphi}_k)^* &= \ln \mathrm{Dir}(\boldsymbol{\varphi}_k|\boldsymbol{\beta}) + \sum_{d=1}^{D}\sum_{n=1}^{N_d} \mathbb{E}_{q(z_{dn})}\left[\delta(z_{dn}=k)\ln \varphi_{kw_{dn}}\right] + \mathrm{const} \\
&= \left(\sum_{v=1}^{V} \beta_v - 1 \ln \varphi_{kv}\right) + \sum_{d=1}^{D}\sum_{n=1}^{N_d} q(z_{dn}=k)\ln \varphi_{kw_{dn}} + \mathrm{const} \\
&= \sum_{v=1}^{V}\left(\beta_v - 1 + \sum_{d=1}^{D}\sum_{n=1}^{N_d} q(z_{dn}=k)\delta(w_{dn}=v)\right)\ln \varphi_{kv} + \mathrm{const} \\
&= \sum_{v=1}^{V}\left(\beta_v + \bar{C}^k_{(\cdot),v} - 1\right)\ln \varphi_{kv} + \mathrm{const} \\
&= \ln \mathrm{Dir}\left(\boldsymbol{\varphi}_k|\boldsymbol{\beta} + \overline{\boldsymbol{C}}^k_{(\cdot)}\right)
\end{aligned}
$$

where

$$
\bar{C}^k_{(\cdot),v} = \sum_{d=1}^{D}\sum_{n=1}^{N_d} q(z_{dn}=k)\delta(w_{dn}=v)
$$

$$
\overline{\boldsymbol{C}}^k_{(\cdot)} = \left[\bar{C}^k_{(\cdot),1}, \bar{C}^k_{(\cdot),2}, \dots, \bar{C}^k_{(\cdot),V}\right]
$$

$$
q(\boldsymbol{\varphi}_k)^* = \mathrm{Dir}\left(\boldsymbol{\varphi}_k|\boldsymbol{\beta} + \overline{\boldsymbol{C}}^k_{(\cdot)}\right) = \mathrm{Dir}(\boldsymbol{\varphi}_k|\boldsymbol{\beta}^*_k)
$$

Practically, the update means to calculate vector of parameters $\boldsymbol{\beta}^*_k = \boldsymbol{\beta} + \overline{\boldsymbol{C}}^k_{(\cdot)}$ for each k

VB update for $q(z_{dn})$

$$\begin{aligned}\ln q(z_{dn})^* &= \mathbb{E}_{q(\boldsymbol{\Theta},\boldsymbol{\Phi})}[\ln P(z_{dn}|\boldsymbol{\theta}_d) + \ln P(w_{dn}|\boldsymbol{\Phi}, z_{dn})] + \mathrm{const} \\ &= \mathbb{E}_{q(\boldsymbol{\theta}_d)}[\ln P(z_{dn}|\boldsymbol{\theta}_d)] + \mathbb{E}_{q(\boldsymbol{\varphi}_k)}[\ln P(w_{dn}|\boldsymbol{\Phi}, z_{dn})] + \mathrm{const} \\ &= \mathbb{E}_{q(\boldsymbol{\theta}_d)}\left[\ln \theta_{dz_{dn}}\right] + \mathbb{E}_{q(\boldsymbol{\varphi}_k)}\left[\ln \varphi_{z_{dn}w_{dn}}\right] + \mathrm{const}\end{aligned}$$

For $q(\boldsymbol{\theta}_d)^* = \mathrm{Dir}\left(\boldsymbol{\theta}_d|\boldsymbol{\alpha}_d^*\right)$ and $q(\boldsymbol{\varphi}_k)^* = \mathrm{Dir}\left(\boldsymbol{\varphi}_k|\boldsymbol{\beta}_k^*\right)$

$$\begin{aligned}\ln \rho_{dvk} &= \mathbb{E}_{q(\boldsymbol{\theta}_d)}[\ln \theta_{dk}] + \mathbb{E}_{q(\boldsymbol{\varphi}_k)}[\ln \varphi_{kv}] + \mathrm{const} \\ &= \psi\left(\alpha_{dk}^*\right) - \psi\left(\sum_{k'=1}^{K} \alpha_{dk'}^*\right) + \psi\left(\beta_{kv}^*\right) - \psi\left(\sum_{v'=1}^{V} \beta_{kv'}^*\right) + \mathrm{const}\end{aligned}$$

Therefore, we update

$$q(z_{dn} = k)^* = \pi_{dw_{dn}k}$$

$$\pi_{dvk} = \frac{\rho_{dvk}}{\sum_{k'=1}^{K} \rho_{dvk'}}$$

Practically, the update means to evaluate 3D matrix with elements π_{dvk}

VB inference for LDA using word counts

- Again, as for the GS inference, rather than using sequence of words w_{dn}, we prefer to represent documents by $D \times V$ matrix $\boldsymbol{M}$, with elements M_{dv} counting how many times document d contains word v
- Instead of estimating each $q(z_{dn} = k)$, we directly estimate expected counts $\bar{C}^k_{(\cdot),v}$ and $\bar{C}^k_{d,(\cdot)}$

$$q(z_{dn} = k) = \pi_{dw_{dn}k}$$

$$\bar{C}^k_{(\cdot),v} = \sum_{d=1}^{D} \bar{C}^k_{d,v} = \sum_{d=1}^{D} \sum_{n=1}^{N_d} q(z_{dn} = k)\delta(w_{dn} = v)$$

$$\bar{C}^k_{d,(\cdot)} = \sum_{v=1}^{V} \bar{C}^k_{d,v} = \sum_{n=1}^{N_d} q(z_{dn} = k)$$

$$\bar{C}^k_{d,v} = \sum_{n=1}^{N_d} q(z_{dn} = k)\delta(w_{dn} = v) = \pi_{dvk} \sum_{n=1}^{N_d} \delta(w_{dn} = v) \quad \Longrightarrow \quad \bar{C}^k_{d,v} = \pi_{dvk} M_{dv}$$

ELBO objective to monitor progress

$$\mathcal{L}\big(q(\boldsymbol{Z},\boldsymbol{\Theta},\boldsymbol{\Phi})\big) = \mathbb{E}_{q(\boldsymbol{Z},\boldsymbol{\Theta},\boldsymbol{\Phi})}\left[\ln\frac{p(\boldsymbol{W},\boldsymbol{Z},\boldsymbol{\Theta},\boldsymbol{\Phi})}{q(\boldsymbol{Z},\boldsymbol{\Theta},\boldsymbol{\Phi})}\right]$$

$$= \sum_{d=1}^{D}\sum_{n=1}^{N_d}\mathbb{E}_{q(z_{dn})}\big[\mathbb{E}_{q(\boldsymbol{\theta}_d)}[ln\,P(z_{dn}|\theta_d)] + \mathbb{E}_{q(\boldsymbol{\varphi}_k)}[ln\,P(w_{dn}|\Phi,z_{dn})] - \ln q(z_{dn})\big]$$

$$- \sum_{k=1}^{K}\mathbb{E}_{q(\boldsymbol{\varphi}_k)}\left[\ln\frac{q(\boldsymbol{\varphi}_k)}{p(\boldsymbol{\varphi}_k)}\right] - \sum_{d=1}^{D}\mathbb{E}_{q(\boldsymbol{\theta}_d)}\left[\ln\frac{q(\boldsymbol{\theta}_d)}{p(\boldsymbol{\theta}_d)}\right]$$

$$= \sum_{d=1}^{D}\sum_{n=1}^{N_d}\sum_{k=1}^{K} q(z_{dn}=k)[\ln\rho_{dw_{dn}k} - \ln q(z_{dn}=k)] - \sum_{d=1}^{D} KL\big(q(\boldsymbol{\theta}_d)||p(\boldsymbol{\theta}_d)\big) - \sum_{k=1}^{K} KL\big(q(\boldsymbol{\varphi}_k)||p(\boldsymbol{\varphi}_k)\big)$$

$$= \sum_{d=1}^{D}\sum_{v=1}^{V}\sum_{k=1}^{K}\bar{C}_{d,v}^{k}(\ln\rho_{dvk} - \ln\pi_{dvk}) - \sum_{d=1}^{D} KL\left(\mathrm{Dir}\big(\boldsymbol{\alpha}_d^*\big)||\mathrm{Dir}(\boldsymbol{\alpha})\right) - \sum_{k=1}^{K} KL\left(\mathrm{Dir}\big(\boldsymbol{\beta}_k^*\big)||\mathrm{Dir}(\boldsymbol{\beta})\right)$$

$$KL\big(\mathrm{Dir}(\boldsymbol{\alpha})||\mathrm{Dir}(\boldsymbol{\beta})\big) = \ln\Gamma\left(\sum_{c=1}^{C}\alpha_c\right) - \sum_{c=1}^{C}\ln\Gamma(\alpha_c) - \ln\Gamma\left(\sum_{c=1}^{C}\beta_c\right) + \sum_{c=1}^{C}\ln\Gamma(\beta_c) + \sum_{c=1}^{C}(\alpha_c - \beta_c)\left(\psi(\alpha_c) - \psi\left(\sum_{c'=1}^{C}\alpha_{c'}\right)\right)$$

Efficient ELBO calculation

$$\mathcal{L}\big(q(\boldsymbol{Z},\boldsymbol{\Theta},\boldsymbol{\Phi})\big) = \sum_{d=1}^{D}\sum_{v=1}^{V}\sum_{k=1}^{K}\bar{C}_{d,v}^{k}(\ln\rho_{dvk} - \ln\pi_{dvk}) - \sum_{d=1}^{D} KL\left(\mathrm{Dir}(\boldsymbol{\alpha}_d^*)||\mathrm{Dir}(\boldsymbol{\alpha})\right) - \sum_{k=1}^{K} KL\left(\mathrm{Dir}(\boldsymbol{\beta}_k^*)||\mathrm{Dir}(\boldsymbol{\beta})\right)$$

Right after updating $q(z_{dn})$ (i.e. evaluating the terms π_{dvk}), the red term becomes independent of k since

$$\pi_{dvk} = \frac{\rho_{dvk}}{\sum_{k'=1}^{K}\rho_{dvk'}} \Rightarrow \ln\rho_{dvk} - \ln\pi_{dvk} = \ln\sum_{k'=1}^{K}\rho_{dvk'}$$

Therefore, right after updating $q(z_{dn})$ and before updating any $q(\boldsymbol{\theta}_d)$ or $q(\boldsymbol{\varphi}_k)$, we efficiently calculate ELBO $\mathcal{L}\big(q(\boldsymbol{Z},\boldsymbol{\Theta},\boldsymbol{\Phi})\big)$ as:

$$\mathcal{L}\big(q(\boldsymbol{Z},\boldsymbol{\Theta},\boldsymbol{\Phi})\big) = \sum_{d=1}^{D}\sum_{v=1}^{V} M_{dv}\ln\sum_{k=1}^{K}\rho_{dvk} - \sum_{d=1}^{D} KL\left(\mathrm{Dir}(\boldsymbol{\alpha}_d^*)||\mathrm{Dir}(\boldsymbol{\alpha})\right) - \sum_{k=1}^{K} KL\left(\mathrm{Dir}(\boldsymbol{\beta}_k^*)||\mathrm{Dir}(\boldsymbol{\beta})\right)$$

where we have used $\sum_{k=1}^{K}\bar{C}_{d,v}^{k} = M_{dv}$, and where we can reuse the terms $\sum_{k'=1}^{K}\rho_{dvk'}$ that were just calculated for normalizing the terms π_{dvk}.

VB inference for LDA: Summary

for number of VB iterations

for every $d = 1..D$, $v = 1..V$ and $k = 1..K$

$$\ln \rho_{dvk} = \psi(\alpha^*_{dk}) - \psi\left(\sum_{k'=1}^{K} \alpha^*_{dk'}\right) + \psi(\beta^*_{kv}) - \psi\left(\sum_{v'=1}^{V} \beta^*_{kv'}\right)$$

$$\pi_{dvk} = \frac{\rho_{dvk}}{\sum_{k'=1}^{K} \rho_{dvk'}}$$

$$\bar{C}^k_{d,v} = \pi_{dvk} M_{dv}$$

for $d = 1..D$

$$\boldsymbol{\alpha}^*_d = \boldsymbol{\alpha} + \overline{\boldsymbol{C}}_{d,(\cdot)}$$

for $k = 1..K$

$$\boldsymbol{\beta}^*_k = \boldsymbol{\beta} + \overline{\boldsymbol{C}}^k_{(\cdot)}$$

where

$$\bar{C}^k_{d,(\cdot)} = \sum_{v=1}^{V} \bar{C}^k_{d,v}, \qquad \bar{C}^k_{(\cdot),\boldsymbol{v}} = \sum_{d=1}^{D} \bar{C}^k_{d,v}$$

$$\overline{\boldsymbol{C}}_{d,(\cdot)} = \begin{bmatrix} C^1_{d,(\cdot)} \\ C^2_{d,(\cdot)} \\ \vdots \\ C^K_{d,(\cdot)} \end{bmatrix}, \qquad \overline{\boldsymbol{C}}^k_{(\cdot)} = \begin{bmatrix} C^k_{(\cdot),1} \\ C^k_{(\cdot),2} \\ \vdots \\ C^k_{(\cdot),V} \end{bmatrix}$$

Where $\boldsymbol{\pi}_{dv}$ is evaluated using the currently values of $\boldsymbol{\alpha}^*_d$and $\boldsymbol{\beta}^*_k$in each iteration.

After running a number of VB iterations, $q(\boldsymbol{\theta}_d)^* = \mathrm{Dir}(\boldsymbol{\theta}_d|\boldsymbol{\alpha}^*_d)$ and $q(\boldsymbol{\varphi}_k)^* = \mathrm{Dir}(\boldsymbol{\varphi}_k|\boldsymbol{\beta}^*_k)$ are approximate posteriors for all $d = 1..D$ and $k = 1..K$